

## EXERCISE 7.2: ALGORITMA PERKALIAN MATRIKS STRASSEN

dikerjakan di rumah, sebelum perkuliahan pertemuan 7

**Petunjuk:** Kerjakan secara berkelompok dengan anggota 3 orang. Anda cukup mengerjakan satu topik saja.

### 1 Perkalian Matriks

#### 1. (Perkalian matriks persegi dengan menggunakan matriks blok)

Ingat kembali algoritma perkalian matriks persegi dengan strategi brute-force. Kita tahu bahwa kompleksitas algoritma tersebut adalah  $\mathcal{O}(n^3)$ .

Perkalian dua matriks persegi dapat dilakukan dengan memecah matriks menjadi matriks blok berukuran  $2 \times 2$ , sebagaimana yang ditampilkan pada ilustrasi berikut.

Misalkan  $A$  dan  $B$  adalah matriks yang berukuran  $n \times n$ . Misalkan  $C = A \times B$ . Matriks  $A$  dan  $B$  masing-masing dipartisi menjadi empat submatriks dengan ukuran  $\frac{n}{2} \times \frac{n}{2}$ .

$$\begin{array}{ccc} \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix} & \times & \begin{bmatrix} B_{11} & B_{12} \\ B_{21} & B_{22} \end{bmatrix} & = & \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \\ A & & B & & C \end{array}$$

$C$  dapat dinyatakan sebagai matriks blok berukuran  $2 \times 2$  yang juga berukuran  $\frac{n}{2} \times \frac{n}{2}$ . Komponen matriks  $C$  dapat dihitung sebagai berikut:

- $C_{11} = A_{11} \cdot B_{11} + A_{12} \cdot B_{21}$
- $C_{12} = A_{11} \cdot B_{12} + A_{12} \cdot B_{22}$
- $C_{21} = A_{21} \cdot B_{11} + A_{22} \cdot B_{21}$
- $C_{22} = A_{21} \cdot B_{12} + A_{22} \cdot B_{22}$

Untuk memperjelas pemecahan matriks persegi menjadi matriks blok, perhatikan contoh berikut.

**Contoh 1.1.** Matriks bujur sangkar dapat dipecah sebagai berikut:

$$A = \begin{bmatrix} 1 & 21 & 15 & 7 \\ 11 & 3 & 10 & 31 \\ 52 & 31 & 2 & 17 \\ 2 & 9 & 23 & 3 \end{bmatrix} \quad \begin{array}{cc} A_{11} = \begin{bmatrix} 1 & 21 \\ 11 & 3 \end{bmatrix} & A_{12} = \begin{bmatrix} 15 & 7 \\ 10 & 31 \end{bmatrix} \\ A_{21} = \begin{bmatrix} 52 & 31 \\ 2 & 9 \end{bmatrix} & A_{22} = \begin{bmatrix} 2 & 17 \\ 23 & 3 \end{bmatrix} \end{array}$$

2. Cermati dan kaitkan pseudocode berikut dengan penjelasan di atas. Jelaskan proses yang terjadi pada algoritma berikut.

---

**Algorithm 1** Matrix multiplication

---

```

1: procedure MMUL( $A, B$ : matrices,  $n$ : integer)
2:   if  $n = 1$  then
3:     return  $A * B$ 
4:   else
5:     SPLIT( $A$ )
6:     SPLIT( $B$ )
7:      $C_{11} \leftarrow$  MSUM(MMUL( $A_{11}, B_{11}, \frac{n}{2}$ ), MMUL( $A_{12}, B_{21}, \frac{n}{2}$ ))
8:      $C_{12} \leftarrow$  MSUM(MMUL( $A_{11}, B_{12}, \frac{n}{2}$ ), MMUL( $A_{12}, B_{22}, \frac{n}{2}$ ))
9:      $C_{21} \leftarrow$  MSUM(MMUL( $A_{21}, B_{11}, \frac{n}{2}$ ), MMUL( $A_{22}, B_{21}, \frac{n}{2}$ ))
10:     $C_{22} \leftarrow$  MSUM(MMUL( $A_{21}, B_{12}, \frac{n}{2}$ ), MMUL( $A_{22}, B_{22}, \frac{n}{2}$ ))
11:   end if
12:   return  $C$ 
13: end procedure

```

▷ The matrices are of size  $1 \times 1$   
▷ Scalar multiplication

▷ SPLIT adalah prosedur untuk memecah matriks menjadi matriks blok berukuran  $2 \times 2$ , dengan setiap blok berukuran  $\frac{n}{2} \times \frac{n}{2}$

▷  $C$  is the union of  $C_{11}, C_{12}, C_{21}, C_{22}$

---

Prosedur MSUM yang digunakan dalam MMUL adalah sebagai berikut.

---

**Algorithm 2** Sum of two matrices

---

```

1: procedure MSUM( $A, B$ : matrices,  $n$ : integer)
2:   for  $i \leftarrow 1$  to  $n$  do
3:     for  $j \leftarrow 1$  to  $n$  do
4:        $C[i, j] \leftarrow A[i, j] + B[i, j]$ 
5:     end for
6:   end for
7: end procedure

```

---

3. Berapakah kompleksitas waktu algoritma perkalian matriks persegi tersebut? Apakah ada peningkatan dibandingkan dengan strategi brute-force?  
(Hint: Gunakan Teorema Master)

## 2 Algoritma perkalian matriks Strassen

1. (Analisis algoritma perkalian matriks dengan pemecahan menjadi matriks blok)

Jika kalkulasi Anda benar, Anda akan menemukan bahwa tidak ada perbedaan yang signifikan pada kompleksitas waktu algoritma di atas dibandingkan dengan strategi brute-force.

Ide Volker Strassen adalah untuk mengurangi banyaknya 'perkalian' dalam prosedur. Karena biaya 'perkalian' lebih 'mahal' daripada 'penambahan' (lihat [https://www.wikiwand.com/en/Computational\\_complexity\\_of\\_mathematical\\_operations](https://www.wikiwand.com/en/Computational_complexity_of_mathematical_operations)).

Operasi yang terjadi pada algoritma sebelumnya terdiri dari 8 perkalian dan 4 penjumlahan:

- $C_{11} = A_{11} \cdot B_{11} + A_{12} \cdot B_{21}$
- $C_{12} = A_{11} \cdot B_{12} + A_{12} \cdot B_{22}$
- $C_{21} = A_{21} \cdot B_{11} + A_{22} \cdot B_{21}$
- $C_{22} = A_{21} \cdot B_{22} + A_{22} \cdot B_{22}$

Strassen memodifikasi persamaan di atas untuk menguranginya menjadi 7 perkalian tetapi dengan konsekuensi terdapat lebih banyak penjumlahan. Dapatkah Anda memperkirakan mengapa teknik ini berpotensi memperbaiki kompleksitas waktunya?

## 2. (Modifikasi oleh Strassen)

Modifikasi perkalian matriks oleh Strassen adalah sebagai berikut:

- $M_1 = (A_{11} - A_{22})(B_{21} + B_{22})$
- $M_2 = (A_{11} + A_{22})(B_{11} + B_{22})$
- $M_3 = (A_{11} - A_{21})(B_{11} + B_{12})$
- $M_4 = (A_{11} + A_{12})B_{22}$
- $M_5 = A_{11}(B_{12} - B_{22})$
- $M_6 = A_{22}(B_{21} - B_{11})$
- $M_7 = (A_{21} + A_{22})B_{11}$

Sehingga:

- $C_{11} = M_1 + M_2 - M_4 + M_6$
- $C_{12} = M_4 + M_5$
- $C_{21} = M_6 + M_7$
- $C_{22} = M_2 - M_3 + M_5 - M_7$

Dalam pseudocode, teknik ini dapat dituliskan sebagai berikut.

---

### Algorithm 3 Matrix multiplication

---

```

1: procedure STRASSEN( $A, B$ : matrices,  $n$ : integer)
2:   if  $n = 1$  then return  $A * B$  ▷ Scalar multiplication
3:   else
4:     SPLIT( $A$ )
5:     SPLIT( $B$ )
6:      $M_1 \leftarrow$  STRASSEN( $A_{12} - A_{22}, B_{21} + B_{22}, \frac{n}{2}$ )
7:      $M_2 \leftarrow$  STRASSEN( $A_{11} + A_{22}, B_{11} + B_{22}, \frac{n}{2}$ )
8:      $M_3 \leftarrow$  STRASSEN( $A_{11} - A_{21}, B_{11} + B_{12}, \frac{n}{2}$ )
9:      $M_4 \leftarrow$  STRASSEN( $A_{11} + A_{12}, B_{22}, \frac{n}{2}$ )
10:     $M_5 \leftarrow$  STRASSEN( $A_{11}, B_{12} - B_{22}, \frac{n}{2}$ )
11:     $M_6 \leftarrow$  STRASSEN( $A_{22}, B_{21} - B_{11}, \frac{n}{2}$ )
12:     $M_7 \leftarrow$  STRASSEN( $A_{21} + A_{22}, B_{11}, \frac{n}{2}$ )
13:     $C_{11} \leftarrow M_1 + M_2 - M_4 + M_6$ 
14:     $C_{12} \leftarrow M_4 + M_5$ 
15:     $C_{21} \leftarrow M_6 + M_7$ 
16:     $C_{22} \leftarrow M_2 - M_3 + M_5 - M_7$ 
17:   end if
18:   return  $C$  ▷  $C$  is the union of  $C_{11}, C_{12}, C_{21}, C_{22}$ 
19: end procedure

```

---

Cermati dan analisis proses yang terjadi pada algoritma di atas. Jelaskan!

## 3. (Kompleksitas waktu algoritma Strassen)

Gunakan Teorema Master untuk menganalisis kompleksitas waktu algoritma Strassen di atas. apakah ada perbedaan dengan hasil yang diperoleh dibandingkan dengan algoritma perkalian matriks yang sebelumnya?